



下载APP



22 | 扩展性设计，看透业务的本质

2020-12-14 乔新亮

乔新亮的CTO成长复盘

[进入课程 >](#)



讲述：乔新亮

时长 22:20 大小 20.47M



你好，我是乔新亮。

这一讲，我想和你聊聊，如何做好扩展性设计。

说到扩展性设计，可能你的第一反应是业务拆分、集群扩容等等。说得没错，这些都能增强系统的扩展性，但仅仅局限于架构和技术层面。我的下属经常兴奋地向我描述，说他实现了一个非常厉害的、高性能、高可扩展性的系统。我的回答经常是，你说的都对，然后呢？



这个问题背后隐含的意思是：对于一名追求成长的技术人来说，只从技术维度思考问题是不够的。这只能让你胜任目前的工作，但不能让你变得卓越。

一名追求卓越的技术人，应该学会思考：我的工作是如何成就业务的，我的产品如何让用户变得更卓越？如果对业务、对用户没有帮助，做再多的技术工作都是无用功。

在职业生涯的早期，每一个工程师都会因为一些技术上的进步被领导夸奖，这很正常。因为在那时，做好基础的代码编写工作，就是你的主要任务。

但如果三年、五年后，你仍然将全部注意力放在技术细节上，就要小心了。比如，很多读者都在专栏下方向我提问，其中一些问题是非常相似的：乔老师，我是一名工作了 7/10/15 年的技术总监，现在感觉非常焦虑，应该怎么突破职业瓶颈，怎么继续成长呢？

如果要将我的回答总结为一句话，我觉得应该是：**让自己变得专业，专业才能成就卓越。**

这个专业不单是指技术越来越厉害了、写代码越来越快了。更重要的是，你在架构设计方面是专业的、在团队管理方面是专业的、在业务发展方面也是专业的。因为专业所以做出了好的产品，通过产品成就了用户，因此也帮助公司业务取得了成功。

回归到今天的主题——「扩展性设计」上，我们同样要考虑：如何做扩展性设计才是专业的？一定不是只会扩展集群，提一些服务器采购需求。

扩展性设计，是为了支撑业务快速发展而出现的概念，目的是保证在企业发展的不同时期，在业务复杂度相近的情况下，业务上线所需要的研发时间不会大幅增加，甚至是基本不变的。本质上，所谓扩展性设计，就是在面向业务的不确定性做设计。

因此，要想做好扩展性设计，设计者要具备企业发展的全局视角，从业务发展的角度出发，倒推出 IT 建设的整个链条，再针对链条中的某个节点，针对性地推进设计工作。

听起来是个很“宏大”的工程，有点让人望而却步。但别害怕，接下来，我们就一步步将其拆解，学习如何做好企业级的扩展性设计。

一条重要的前置思考脉络

要做好企业级的扩展性设计，意味着你要强迫自己像 CEO 一样思考，首先理顺一条重要的前置思考脉络。

如同前文所讲，无论是高可用、高性能还是今天聊到的扩展性设计，出发点都应该是业务发展，所以我们首先可以明确，**这条脉络的第一个节点，是公司的年度或季度业务发展目标。**

那么，企业用什么形式来支撑业务的发展和增长呢？如果你认真听了前面的课程，此刻心中应该会有答案。没错，答案是产品。所以**第二个节点，是企业的产品建设。**

产品，本质上是一种顶层设计，底层要由众多应用 / 业务组件支撑。所以**第三个节点，是企业的应用架构设计。**

而应用架构又由众多技术组件在底层提供基础能力支撑，所以**第四个节点，是企业级技术架构设计。**

以上四个节点，最终都是由人来完成的。所以我们还要考虑组织的人才梯队建设，包括 A/B 岗配置、同赛道竞争机制等。一旦将团队纳入设计，就要考虑协同效率的问题，不能因为业务增长、团队增长而引入协同问题。这两点，则全部属于团队管理范畴的问题。

其实没有那么复杂，对吧？我们来梳理一下，要提升扩展性，需要在以下四个层面进行设计，分别是：

1. 公司的年度 / 季度业务发展目标；
2. 企业级产品建设；
3. 企业级应用架构设计；
4. 企业级技术架构设计。

同时，我们也要将团队管理内容纳入考量，分别是人才梯队建设和提升协同效率。

如果用云计算领域的概念来比喻，那么「产品建设」就像 SaaS、「应用架构设计」就像 PaaS、「技术架构设计」就像 IaaS/PaaS，三者共同固化为企业的核心能力，在团队管理方法的辅助下，实现业务可持续的向前发展。

这四点就像一张“寻宝地图”，下面我们来分析一下，如何通过这张“地图”，实践扩展性设计。

保证企业级年度 / 季度业务发展目标实现聚焦

制定「企业级年度 / 季度业务发展目标」，需要在公司战略确定、商业模式确定的情况下，充分考虑市场竞争情况，确定目标的优先级，明确每个季度的工作目标。

此时，我们主要考虑节奏问题。这里的“节奏”是什么意思呢？从企业发展的角度讲，一切能力都需要提升，但顺序不一样，节奏就不一样，效果上的差别也非常大。所以，做好战略规划很重要，要分析各战略步骤的依赖关系、重要性、紧急程度，最后确定执行顺序。

所谓“扩展性设计”，通常是当大环境或创始团队的认知出现变化时，公司需要重新调整业务目标，各团队也要配合调整。具体到每个季度的目标，则一定要清晰、聚焦、上下对齐。关于战略执行效率的问题，我们在“[🔗 管理三板斧](#)”部分进行过讲解，可以参考一下。

此外，初 / 中级管理者很难像 CEO 一样透彻理解业务目标，但要尽力去理解、去沟通，每个季度要主动和公司目标进行对齐。认知越透彻，对自己的工作和成长越有帮助。

用业务目标指导产品建设

确定了业务目标后，在规划产品建设时，团队可以通过三步做好扩展性设计：

1. 通过架构思维，将产品拆解为一个个功能模块；
2. 针对每个模块，用穷举法思考其他扩展可能；
3. 以 ROI 为出发点，对所有可能进行收敛，最终确定要落实的扩展性设计。

对于步骤一，我们就不再展开详谈了，可以参考 [🔗 《架构设计，专业分工和协作精神的体现》](#) 这一讲的具体内容；对于步骤二，相关负责人既可以根据市场趋势来穷举，也可以根据相关竞品来穷举，其实更像一场头脑风暴。

关键在于步骤三，如果收敛得不好，会让产品变得臃肿，变成过度设计；如果收敛得太过，又会使产品缺乏扩展性。如何准确计算 ROI，是一个大学问。而计算 ROI 的重任，一般都会落在产品经理的头上。

ROI 的意思是投资回报率，也叫做投入产出比，常用的计算公式有五种以上。在技术管理领域，我们可以简单将 ROI 理解为“收益 / 投入”。

业务侧人员一般重点关注工作收益，不关心工作投入；技术侧人员则非常重视工作投入，不关心工作受益；产品经理则要学会综合考虑。

无论是业务侧、技术侧还是产品侧，这三方人员的业务思维越好，产品需求的收敛工作就完成得越快、越好。前面我们讲过，IT 团队的每个人都要懂业务，其意义就在于此；相应的，业务人员也需要培育产品思维，这样才能最终实现公司的业务 IT 一体化。

当然，产品建设一方面从业务侧收集需求，完善产品；另一方面，优秀的产品经理也会帮助业务筛选客户，确定产品方向。二者循环往复，互为补充。

举个例子，一个生鲜类 To B 产品的报价系统，通常需要和对标对象做价格比对，并做相应调整。通常，一个没有扩展性思维的团队，接了需求就会直接开始写代码，反正也没什么难度。

但一个有扩展性思维的团队会思考：在北京需要与对标对象 A 比对价格，在福州可能就是与对象 B 比对价格了，不同的地区可能会对系统功能有不同的需求。

进而，我们可以将一个价格比对产品的功能模块进行抽象，包含：对标对象设置、对标商品映射、价格设定规则（上浮或者下调）、对标周期等，这样就可以满足全国各个区域的任何价格比对需求，支持业务的快速发展。

所以，产品能否具备高扩展性，是与产品经理的业务思维和抽象能力密切相关的。同时，如果产品设计做得好，应用架构的设计工作就会简单很多。

讲完了业务目标和产品设计，也就来到了关于扩展性设计的第三个节点：做好企业的应用架构设计。这里，我们需要重点区分两个概念：架构设计的确定性与不确定性。

扩展性设计，为“不确定性”而设计

笼统地讲，任何业务架构都存在确定性和不确定性的部分，但二者并不是恒定不变的。所谓的确定性部分，可能无法适应业务的演进和发展，因此出现大量改动；而所谓的不确定性部分，也可能随着时间的推移，逐渐固化为产品能力，变成设计内的确定性内容。

企业级应用架构设计可以包含：

1. 交易体系；
2. 协同体系；
3. 监控指挥体系；
4. 生产体系。

这是做拆分的基本思想，无论在顶层还是底层，都非常适用，需要做的只是不断进行拆分。经过这些年的工作实践，我觉得，无论是组织管理、架构设计还是产品设计，很多时候就像“套娃”。刚开始做企业架构规划时，我觉得非常难以入手，彻底掌握后，我发现这和做系统架构设计简直一摸一样。

对于生鲜行业的 To B 产品来说，查询、下单、发货、签收，是始终不变的业务流程，属于确定性部分。这部分的实现归属交易体系。**交易体系处理确定性问题，一般采用 SOA 架构。**

但在实际的业务场景中，意外会经常出现。比如，客户下单购买了 500 斤白菜，但在发货时，仓储系统却显示备货不足，无法正常发货。

此时，产品预设的服务流程就被打破了。企业一般会指派客服人员与客户联系，尝试着商量一下：“不好意思，我们的白菜只剩 300 斤了，剩下的 200 斤给您换成芹菜吧，芹菜多吃呀！”

这个时候，我们就需要让采购人员、销售支持、销售人员联系客户，进行充分沟通，确定最终的商品品类及数量。这个流程属于协同问题，充满不确定性，也就是说，相比交易体系，更有可能随着公司的管理优化而进行变化。**协同体系用于处理不确定性问题，一般采用 EDA 架构，且要和交易体系进行集成。**

所谓「不确定性」部分，其实正是业务架构做扩展性设计的核心。交易体系和协同体系的分离，等同于分离了业务的确定性和不确定性部分，因此非常有利于业务功能的扩展。

分离不代表完全无关，交易体系和协同体系的集成点，我称之为 CP (control point) 。一般来说，任何一个 CP 都要被监控、分析、控制，这就是企业的监控指挥体系。监控指

挥体系和公司的管理密切相关，往往是公司数据化管理的重要抓手。

监控指挥体系可以分为监控、分析、洞察、控制等几大功能，大数据和 AI 部分的技术内容也在这个体系中。监控体系解决了公司的很多问题，但怎样保证产品高效地迭代和优化呢？这就是生产体系要解决的问题。

生产体系解决公司研发管理地速度问题。我的观点是，技术管理者要学会引入“流水线”式设计思想，尝试极大简化开发人员、测试人员的工作复杂度。如果将研发的整体流程看作一条“流水线”，代码开发完成后，工作要自动流转，正常、异常都要自动流转至对应人员进行处理，这是有关 CI、CD、CO 的内容，我们不展开详解，但它非常重要。

做好了应用架构层面的扩展性设计，我们也就来到了第四个节点——企业级技术架构设计。

技术架构设计设计的关键在于：**不要重复造轮子，至少不要在公司层级造轮子。**“局部最优，整体很差”的情况，很多时候都是因为重复造轮子导致的。

所以，要实现技术架构体系中的各个技术平台，可以通过两步完成：

1. 自研；
2. 购买相应的套装软件或云服务。

对于非核心系统，在需求较为匹配的情况下，建议选择购买套装软件或云服务；对于高度定制化的企业核心系统，也就是在核心价值链上提供服务的系统，则建议自研。

是不是很简单？到这里，我们就将扩展性设计的四个关键节点全部分析完毕了。

其实真正的扩展性设计，往往与单一维度的技术问题无关，也与某个人的架构设计能力关系不大——扩展性设计，是团队整体认知、博弈与决策的结果。

这意味着，如果你想在一家公司内，充分践行以上设计思想，需要锻炼并掌握一定的表达技巧，与领导、同事、下属保持充分的沟通，更与组织管理能力息息相关，要多治治“腼腆内向”、“不善沟通”等技术人“职业病”。

结语

这一讲，我们聊了聊如何进行扩展性设计。

可以说，**真正优秀的扩展性设计，建立在看透业务本质的基础上，面向不确定性，但从不确定性中寻找确定性。**

这里需要注意，越是行之有效的方法，越不会太过复杂。比如说，研发速度快，对于产品或架构而言，也是一种扩展性——我们没做过什么扩展性设计，但团队的研发速度很快：今天发现需求，明天就能上线。

这样靠谱吗？当然靠谱，就像我们之前说的，“天下武功，唯快不破”。只是长期看来，唯有体系化的思维和解决方案，才能真正从根本上解决问题。

如果你有问题，欢迎在评论区向我提问；如果你觉得有帮助，也欢迎分享文章到朋友圈，让更多人参与进来，共同学习。

我们下一讲再见！

提建议

训练营推荐

产品经理训练营

成为有大局观、懂用户、能拿结果的产品经理

邱岳 十年资深产品经理，无码科技联合创始人

立抢优惠名额 

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 21 | 高性能设计，一切都围绕着契约精神

下一篇 23 | 考虑限制，让自己的产品不入险地

精选留言 (6)

 写留言

Weehua

2020-12-17

了解业务，熟悉业务，才能做出最适合的扩展性设计。以终为始，永远都从公司的业务目标出发来进行技术相关的架构和设计。这种业务思维和产品思维，真的是我们技术人员需要加强学习的，也是上台阶的重要能力。虽然越往后学越感觉困难，但仔细研读，每次都有提高认知的感觉，这种感觉太棒了，感谢乔老师，这种才是传道受业解惑。

...

展开 

作者回复: 你好，Weehua

我管理的团队，做成什么样，产品经理负责，不是业务人员决定的。

产品经理要考虑业务发展，变化，但是业务的问题要解决，如何做业务没有权力做决定。

 1 1

**天天向上**

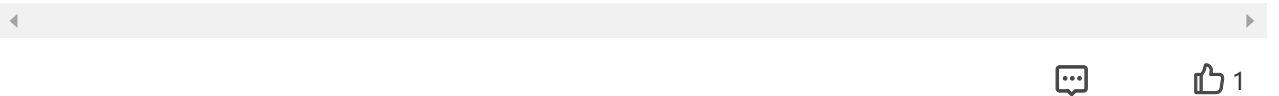
2020-12-17

CP有点抽象没理解 能举个例子说明下吗？
以及这些CP需要监控 关注的核心要点？

作者回复: 你好，天天向上

比如一个销售订单进入到仓库作业后，发现需要的5瓶矿泉水没有了，而这个是在下单的时候检查库存没有问题的。可能因为是库存管理不准，可能是破损盘亏了，但不管是怎样，这里少了矿泉水，导致履约出现问题，会触发紧急采购。

在触发紧急采购的地方，就是一个需要监控、控制点，为什么呢，这个地方的触发一定会导致用户体验下降，采购成本上升，要监控数据，通过管理降低这里触发，甚至是消灭。向管理要效益。

**Lemon**

2020-12-14

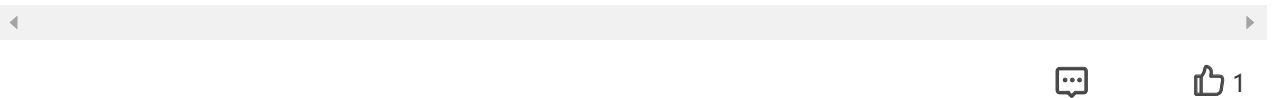
本章内容对于当前个人水平而言太过遥远，但还是能从中得到感悟，等手头忙完了一定再回顾一遍。现阶段来看，拓展性就是为了让产品能用较低的代价快速响应市场，暂时粗浅的将拓展性设计的步骤理解为：

1. 公司的年度 / 季度业务发展目标； ...

展开 ∨

作者回复: 你好，Lemon

这种学习态度，真的要点赞，加油

**李**

2020-12-14

老师，从哪些方面可以锻炼养成体系化思维能力，并且能一下抓住重点。其实看老师课程，受益匪浅，并不仅仅是看老师内容，主要是看老师怎么去讲的，怎么去分析的，思考老师为什么要这么取讲，我觉得这个背后的思维逻辑更重要

展开 ∨

作者回复: 你好，李

遇到问题多思考，多思考背后的逻辑，哪些是稳定的，哪些是变化的。

学习一些对应的知识，培养自己的逻辑思维，思辨能力。可以去看看比如麦肯锡的金字塔，还有逻辑思维的书籍，我的公众号：乔新亮，一本好书 里面推荐过，可以去看看。



剑八

2020-12-20

扩展性设计方法论：

1，从业务出发，理解业务本质及发展趋势

识别变与不变

如交易中流程下单，支付，发货，签收是不变的，具体支付方式可能变化

2，顶层设计，最终产品是？核心功能是？ ...

展开 ∨

作者回复: 你好，剑八

加油，总结的好。



spark

2020-12-14

从对价值假设的验证出发，Build-Measure-Learn，首先不要用固化和简单思维看问题，直面变化给我们的挑战，这是思考可扩展性的起点。

我们或许可以用OKR工具去管理我们的目标，迭代我们的业务和组织成熟度。

最后从代码实现的角度认知可扩展性：识别我们的核心能力，稳定我们的核心能力。用开闭原则设计我们的新需求，就算需求带来可扩展性方面大的挑战，只要我们不修改旧代...

展开 ∨

作者回复: 你好，spark

成长性思维，棒

